

# Języki skryptowe - opis przedmiotu

## Informacje ogólne

Nazwa przedmiotu	Języki skryptowe
Kod przedmiotu	11.3-WE-INFP-JęzSkryp
Wydział	<a href="#">Wydział Informatyki, Elektrotechniki i Automatyki</a>
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2019/2020

## Informacje o przedmiocie

Semestr	4
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	• dr inż. Piotr Witczak

## Formy zajęć

Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	18	1,2	Egzamin
Laboratorium	30	2	18	1,2	Zaliczenie na ocenę

## Cel przedmiotu

- Zapoznanie studentów z językami skryptowymi (w szczególności z językiem Python)
- Ukształtowanie umiejętności praktycznego zastosowania języków skryptowych
- Ukształtowanie umiejętności wykorzystywania gotowych bibliotek i frameworków

## Wymagania wstępne

- Podstawy programowania
- Programowanie obiektowe

## Zakres tematyczny

- paradygmat języków skryptowych i interpretowanych
- różnice pomiędzy językami interpretowanymi a kompilowanymi (na przykładzie języków Python i C++)
- zastosowanie języków skryptowych do tworzenia stron www
- wykorzystanie języków skryptowych w tworzeniu nowoczesnego oprogramowania
- wykorzystanie języków skryptowych do obliczeń inteligentnych

## Metody kształcenia

- wykład: wykład konwencjonalny
- laboratorium: ćwiczenia lab., praca w grupach, programowanie w parach,

## Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Ma wiedzę na temat trendów rozwojowych dyscypliny oraz potrzeby usprawniania codziennych czynności informatycznych poprzez stosowanie języków skryptowych	<ul style="list-style-type: none"><li>• <a href="#">K_W05</a></li><li>• <a href="#">K_W09</a></li><li>• <a href="#">K_W19</a></li></ul>	<ul style="list-style-type: none"><li>• egzamin - ustny, opisowy, testowy i inne</li></ul>	<ul style="list-style-type: none"><li>• Wykład</li></ul>
Rozumie potrzebę stosowania języków skryptowych	<ul style="list-style-type: none"><li>• <a href="#">K_W09</a></li><li>• <a href="#">K_W13</a></li></ul>	<ul style="list-style-type: none"><li>• dyskusja</li></ul>	<ul style="list-style-type: none"><li>• Wykład</li></ul>
Potrąfi zastosować techniki skryptowe do realizacji zadania będącego częścią większego projektu lub systemu informatycznego	<ul style="list-style-type: none"><li>• <a href="#">K_W09</a></li><li>• <a href="#">K_W19</a></li><li>• <a href="#">K_U15</a></li><li>• <a href="#">K_U18</a></li></ul>	<ul style="list-style-type: none"><li>• obserwacje i ocena umiejętności praktycznych studenta</li></ul>	<ul style="list-style-type: none"><li>• Laboratorium</li></ul>
Dokonuje wyboru właściwych narzędzi oraz rozwiązań skryptowych w zależności od wymagań projektowych	<ul style="list-style-type: none"><li>• <a href="#">K_W09</a></li><li>• <a href="#">K_K09</a></li></ul>	<ul style="list-style-type: none"><li>• obserwacje i ocena umiejętności praktycznych studenta</li></ul>	<ul style="list-style-type: none"><li>• Laboratorium</li></ul>

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi zastosować języki skryptowe do rozwiązania problemu naukowego oraz inżynierskiego	<ul style="list-style-type: none"> <li>• <a href="#">K_U01</a></li> <li>• <a href="#">K_U15</a></li> <li>• <a href="#">K_U18</a></li> <li>• <a href="#">K_K09</a></li> </ul>	<ul style="list-style-type: none"> <li>• obserwacje i ocena umiejętności praktycznych studenta</li> </ul>	<ul style="list-style-type: none"> <li>• Laboratorium</li> </ul>
Rozumie paradygmat języków skryptowych oraz interpretowanych oraz różnice w stosunku do języków kompilowanych	<ul style="list-style-type: none"> <li>• <a href="#">K_W05</a></li> <li>• <a href="#">K_W09</a></li> </ul>	<ul style="list-style-type: none"> <li>• egzamin - ustny, opisowy, testowy i inne</li> </ul>	<ul style="list-style-type: none"> <li>• Wykład</li> </ul>

## Warunki zaliczenia

**Wykład** - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu

**Laboratorium** - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium

**Składowe oceny końcowej** = wykład: 50% + laboratorium: 50%

## Literatura podstawowa

1. Learning Python: Powerful Object-Oriented Programming, Mark Lutz, "O'Reilly Media, Inc.", 2013, 9781449355692
2. Python Crash Course: A Hands-On, Project-Based Introduction to Programming, Eric Matthes, No Starch Press, 2015, ISBN: 9781593276034
3. Mastering TensorFlow 1.x, Armando Fandango, Packt Publishing Ltd, 2018, 9781788297004
4. Beginning Django: Web Application Development and Deployment with Python, Daniel Rubio, Apress, 2017 ISBN: 978148422787

## Literatura uzupełniająca

1. Python Cookbook, Alex Martelli, Anna Ravenscroft, David Ascher, "O'Reilly Media, Inc.", 2005, ISBN: 9780596554743
2. Effective Python: 59 Specific Ways to Write Better Python, Brett Slatkin, Addison-Wesley Professional, 2015, ISBN: 9780134034409
3. Fluent Python: Clear, Concise, and Effective Programming, Luciano Ramalho, "O'Reilly Media, Inc.", 2015, ISBN: 9781491946251
4. Scripting Languages: Automating the Web, Rohit Khare, O'Reilly, 1997, ISBN: 9781565922655
5. Architectural Support for Scripting Languages, Dibakar Gope, University of Wisconsin–Madison, 2017
6. Software Architecture: A Comprehensive Framework and Guide for Practitioners, Oliver Vogel, Ingo Arnold, Arif Chughtai, Timo Kehrer, Springer Science & Business Media, 2011, ISBN: 9783642197369

## Uwagi

Zmodyfikowane przez dr inż. Piotr Witczak (ostatnia modyfikacja: 24-04-2019 10:25)

Wygenerowano automatycznie z systemu SyllabUZ